README FILE

Project Title: AWS VPC and EC2 with Terraform

Project Over-View

This project demonstrates the automation of deploying a basic AWS infrastructure using Terraform, which includes creating:

1. VPC (Virtual Private Cloud)
2. Subnets (Public and Private)
3. Internet Gateway
4. Route Table and Route Table Associations
5. EC2 Instance (with NGINX Web Server)
6. Security Groups

Here is a detailed explanation of each resource and its use in this project:

1. VPC (vpc.tf)

- Virtual Private Cloud (VPC) is the fundamental network structure in AWS. This creates a private network in AWS, isolating your resources from other networks in the AWS cloud.
- CIDR block (10.0.0.0/16): Specifies the IP range for the VPC.
- Private Subnet (aws_subnet.private-subnet): Used for resources that should not directly be accessed from the internet (e.g., backend databases).
- Public Subnet (aws_subnet.public-subnet): Used for resources that need to be accessed from the internet (e.g., web servers).

2. Internet Gateway and Route Table (vpc.tf)

- Internet Gateway: An Internet Gateway is attached to the VPC to allow communication between instances in the VPC and the internet.
- Route Table: Defines how network traffic flows between different resources and outside the VPC.
- The route 0.0.0.0/0 means that all traffic should go through the Internet Gateway to reach the internet.
- Route Table Association: This connects the public subnet to the route table, allowing EC2 instances in the public subnet to access the internet.

3. **EC2 Instance (ec2.tf)**

- EC2 Instance: This is an AWS virtual machine (VM). In this project, the EC2 instance is configured to run a basic NGINX web server.
- AMI: Amazon Machine Image (AMI) specifies the base operating system for the EC2 instance (ami-0327f51db613d7bd2 is a pre-configured Amazon Linux AMI). ==> Instance Type: The EC2 instance is of type t2.micro, which is eligible for the AWS free tier.
- User Data Script: This script installs and starts the NGINX web server when the EC2 instance is launched.
- #!/bin/bash
  sudo yum install nginx -y
  sudo systemctl start nginx
- Security Group: It is associated with the nginx-sg security group that allows HTTP traffic on port 80.

4. **Security Group (security-groups.tf)**

- Security Group: This acts as a virtual firewall for the EC2 instance, controlling inbound and outbound traffic.
- Ingress: This allows HTTP traffic on port 80 from any IP address (0.0.0.0/0).
- Egress: This allows outbound traffic from the EC2 instance to any destination (0.0.0.0/0).
- Security Group: It is associated with the nginx-sg security group that allows HTTP traffic on port 80.

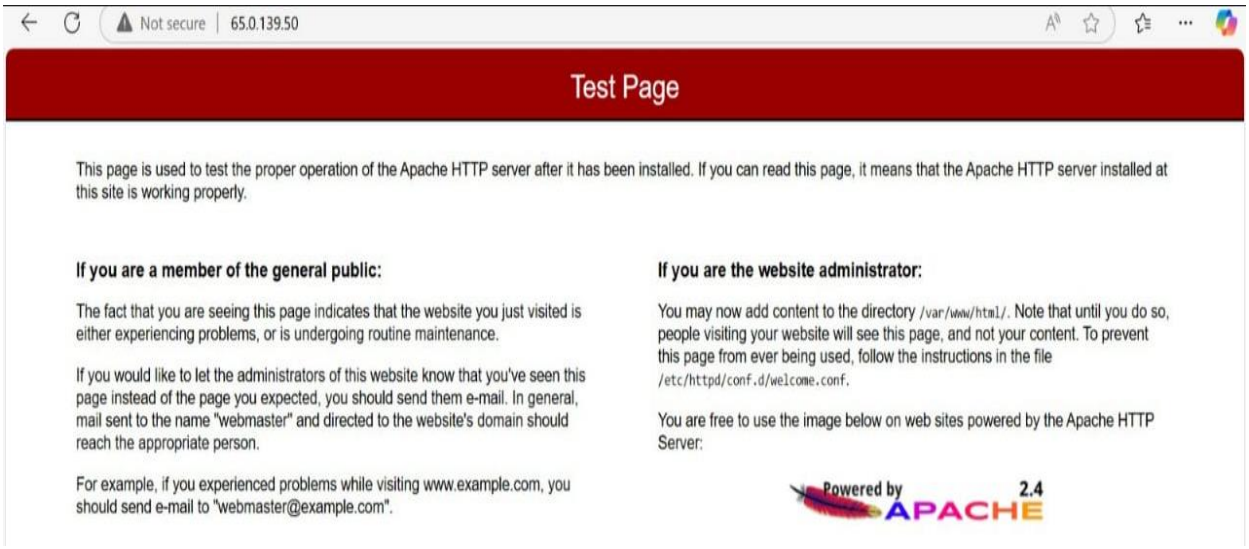
5. **Outputs (outputs.tf)**

- Outputs: The outputs section is used to display useful information after the infrastructure is created.
- instance_public_ip: Displays the public IP address of the EC2 instance, which can be used to access the NGINX server.
- instance_url: A formatted URL (http://<instance_public_ip>) that provides direct access to the NGINX web server.

Project Use Case:

- This project demonstrates the process of:
- Creating a basic network infrastructure in AWS using Terraform, which includes VPC, subnets, internet gateway, and route tables.
- Launching an EC2 instance in the public subnet to serve as a web server using NGINX.
- Securing the EC2 instance using a security group that allows only HTTP traffic on port 80.
- Outputting the EC2 instance's public IP and a URL to access the NGINX web server.
- How to Use This Project:
- Install Terraform: Ensure you have Terraform installed on your machine.
- Configure AWS Credentials: Ensure that your AWS credentials are set up (e.g., using AWS CLI).
- Initialize Terraform: Run terraform init to initialize the working directory containing the Terraform configuration files.
- Apply the Terraform Plan: Run terraform apply to apply the configuration and deploy the infrastructure.
- Access the Web Server: Once the deployment is complete, visit the EC2 instance's public IP or use the output URL to see the default NGINX page.

Output


Not secure | 65.0.139.50
Test Page
This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.
If you are a member of the general public:
The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.
If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.
For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".
If you are the website administrator:
You may now add content to the directory /var/www/html/. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file /etc/httpd/conf.d/welcome.conf.
You are free to use the image below on web sites powered by the Apache HTTP Server:
Powered by
APACHE
2.4